Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 44 (2015) 31 – 41

Procedia Computer Science

2015 Conference on Systems Engineering Research (CSER'15)

Complexity based risk evaluation in engineered systems

Jonathan Fischi*, Roshanak Nichiani

Stevens Institute of Technology, Castle Point on Hudson, Hoboken NJ 07030, USA

Abstract

This work proposes a novel approach to augment program risk management processes by introducing complexity-based risk evaluation which potentially exposes risks that would have otherwise been neglected. The approach is to determine the complexity-based risks by first evaluating system, observer, and behavior entropy models. The suggested metrics created by the models provide the framework for generating complexity-based risks. The results can be used by program management to aid in decision-making processes. The results can also be shared with customers to determine any appropriate short and long-term mitigation actions. The suggested metric are applied to a sample case study. Future work will include impact analysis and mitigation strategies for complexity-based risks.

Peer-review under responsibility of the Stevens Institute of Technology.

Keywords: system complexity; risk management; system design

1. Introduction

Risk assessment of engineered systems designs have been of great interest to program managers as well as all stakeholders of engineered systems. Risks have a future potential consequence based on a root cause. Program management desires visibility into these causes and consequences. Investigations are performed by expert team members and the data is compiled for evaluation of probability and consequences. Since risk assessments are created by individuals based on their rational opinions, they are inherently subjective and vulnerable to biases. A set of risks may even be neglected due to the subjective nature of the process [1]. Where system complexity is concerned, the

* Corresponding author. Tel.: 315-456-1978; fax: +1-201-216-5541.
E-mail address: jfischi@stevens.edu

Peer-review under responsibility of the Stevens Institute of Technology.
doi:10.1016/j.procs.2015.03.044

root causes may not be discovered without a framework to help guide the evaluation. This work proposes a method that augments the existing risk management process to expose complexity-based risks that may otherwise have not been readily apparent.

This paper begins with a literature review on system complexity and risk and a reference definition of complexity is established. The reference complexity definition is used to develop a system, observer, and behavior entropy model using the Goal, Question, Metric (GQM) method. Complexity metrics are derived and evaluated for significant probability of occurrence. Finally, complexity-based risks are compiled and evaluated for probability of occurrence. A case study is described and evaluated using the proposed methodology.

2. Literature review

A definition of complexity, herein referred to as the 'reference definition', is the degree of difficulty in accurately predicting future behavior such that complexity is determined by the system being observed, the capabilities of the observer, and the behavior that the observer is attempting to predict [2] [3]. Peter Erdi indicates that complex systems properties include aspects of circular causality, feedback loops, small changes resulting in out-of-proportion results, and Emergence/unpredictability [4]. Erdi's definition in the context of the reference definition is given in Table 1.

Table 1 - Complexity definition correlation: Erdi [4]

Reference definition	Erdi complexity aspects
Complexity is determined by: The system being observed	System properties that allow for difficult to explain behavior (Circular causality, Feedback loops, logical paradoxes, strange loop)
Complexity is determined by: The capabilities of the observer	System produces responses of scale unexpected by observer (Small changes resulting in out-of-proportion results)
Complexity is determined by: The behavior that the observer is attempting to predict	System has some aspect that is unpredictable / emergent

Sargut and McGrath indicate that predictability is the primary distinction between a complicated system and a complex system. Complex systems are predictable in that they have been designed to continuously adjust, but not predictable in that the system reacts to a host of stimuli [5]. The Sargut and McGrath definition in the context of the reference definition is given in Table 2.

Table 2 - Complexity Definition Correlation: Sargut and McGrath [5]

Reference definition	Sargut and McGrath complexity aspects
Complexity is determined by: The system being observed	Unintended consequences result from simple actions
Complexity is determined by: The capabilities of the observer	Observer has difficulty "making sense of a situation." "Human beings' cognitive limits mean that no manager (observer) can understand all aspects of the business (system)"
Complexity is determined by: The behavior that the observer is attempting to predict	Predictability is the primary distinction between and a complex system and a merely complicated system. "Rare events are more significant than average ones." Interaction of various parts of the system result in unpredicted results

The MIT Engineering Systems Division (ESD) defines a complex system as a "system with components and interconnections, interactions, or interdependencies that are difficult to describe, understand, predict, manage, design, or change" [6]. The MIT ESD definition in the context of the reference definition is given in Table 3.

Table 3 - Complexity Definition Correlation: MIT ESD [6]

Reference definition	MIT ESD complexity aspects
Complexity is determined by: The system being observed	System interdependencies may unintended; dynamic system; many intricately connected components
Complexity is determined by: The capabilities of the observer	Components and interconnections, interactions, or interdependencies the observer finds difficult to describe, understand, predict, manage, design, or change
Complexity is determined by: The behavior that the observer is attempting to predict	The complexity of a system can be quantifiable, but what makes a system appear complicated to the observer is subjective; How complicated the system appears depends on the nature of the interface of the system that is presented to its users.

Similarly, Dodder and Dare of MIT indicate that Complex Adaptive Systems, a category of complex systems have a high degree of difficulty accurately predicting future behavior [7]. Sheard and Mostashari indicate that dynamic complexity includes system behavior and evolving complexity versus time [8]. MITRE created a Systems Engineering Profiler tool to quantify program complexity [9]. The tool objective is to quantify mission evolution, effort scope, effort scape, acquisition environment and stakeholder relationship stability. Nilchiani et al. evaluated risk management techniques and proposed an objective risk management framework [1]. The novel risk management approach identifies sources of complexity in a completely objective manner.

Several research groups participated in the 2011 DARPA META II Complex Systems Design and Analysis (CODA). One of the relevant questions was regarding complexity metrics, the metrics relevance to schedule/cost/reliability, and how useful the metrics are in comparing alternate designs. A participating research group from Boeing Company, Purdue University, and Arizona State University selected a wide range of complexity metrics for evaluation against schedule/cost/reliability data sets from real world programs to create a composite, abstract complexity metric [10]. The team performed a statistical analysis to identify correlations between computed metrics and the observed behavior of cost/schedule/reliability and found strong correlations for particular metrics. The team indicated that comparing complexity metrics between system types would not be advisable. A participating research group from Massachusetts Institute of Technology developed entropy-based metrics and used them to characterize uncertainty in the system development processes [11]. The objective was to quantify complexity in a way that it can be managed during development. This research proposes a statistical method for generating a complexity curve based on entropy and maximum entropy. Another participating research group from United Technologies Corporation Complex Systems Design and Analysis (CODA) produced an architectural design space exploration method. User defined data inputs and iterations on the model determine the best architecture [12].

The Aerospace Corporation developed a complexity analysis for NASA missions which includes complexity index computations for comparisons between programs [13]. The index consists of system design parameters, the estimated complexity of the parameters, and an aggregate complexity for the system. Suh proposed axiomatic design principles to aid in design decision making, particularly for complex systems. Suh proposed a means to tie complexity to the probability of satisfying requirements [14]. The DARPA funded Abstraction Based Complexity Management effort completed by United Technologies Corporation proposes an abstraction based enumeration technique for complexity-based system design. Complexity numbers are created for potential system architectures which are then modeled against design and uncertainty models for determination of the optimal architecture [15].

3. Complexity-based risk evaluation

Certain forms of system complexity are inherently difficult to identify and characterize which makes aspects of complexity prone to be ignored during the course of standard program risk management processes. This section describes creation and evaluation of complexity-based risk assessments using established complexity definitions. This work is not intended to replace the existing risk management processes but to augment it with a mechanism to identify complexity-based risks that would otherwise have been neglected.

3.1. Complexity definition and aspects of complexity

The reference definition of complexity suggested by the literature search is "the degree of difficulty in accurately predicting future behavior" such that complexity is determined by the system being observed, the capabilities of the observer and the behavior that the observer is attempting to predict. The reference definition contains the key aspects of the other definitions found in the literature search (see Table 1, Table 2, and Table 3). This reference definition will be used for subsequent complexity-based risk development in this effort.

Complexity can be of a structural nature. Structural complexity deals with architecture connectivity such as the number of subsystems and the quantity of different types of interfaces. Structural complexity is relatively easy to identify. Dynamic complexity refers to the behavior or responses of the system which can be much more difficult to identify. Dynamic complexity can include adaptive responses, non-linear changes, self-organization and "emergent behavior." Socio-political complexity deals with human aspects like stakeholders and political environments. Although not conclusively demonstrated by this effort, structural complexity is probably more readily identified by standard risk management techniques since it is visually observable in architecture diagrams. For this reason the case study investigated later in this work focuses on dynamic and socio-political complexity-based risk.

3.2. Complexity-based risk investigation

The next step is determining how our understanding of complexity can help the systems engineer evaluate complexity-based risks in a system. The well-established GQM method [16] [17] is used to develop metrics. The Goal Question Metric (GQM) approach was designed as a top-down software measurement approach to assist in project planning, to determine strengths and weaknesses of software or process, to determine benefits of refinement techniques, and to evaluate product/process quality. Using a top-down approach is important since the number of possible metrics viewed from the 'bottom', or at the implementation level, is very large. The only way to properly represent the desired measurements is by understanding the context in which they are needed. The principle of GQM is that project goals (the conceptual level) must drive questions (the operational level) which are represented by metrics (the quantitative level) (see Figure 1). Metrics are a system of measurement, and in our case the metrics will be used as potential indicators of risk.

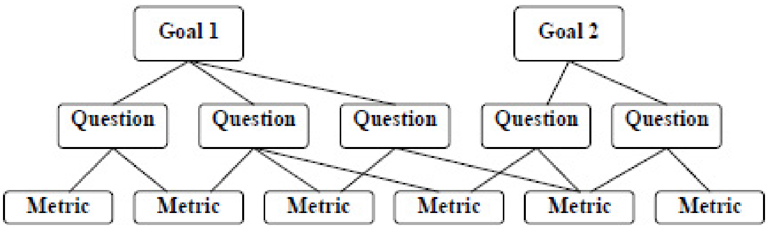

Figure 1 - GQM hierarchical structure [17]

The goals in Table 4 are determined from the reference definition that complexity is the degree of difficulty in accurately predicting future behavior such that complexity is determined by

- the system being observed
- the capabilities of the observer
- the behavior that the observer is attempting to predict

For the GQM Model, the following definitions will apply:

- System Mission includes requirements, uses, scale, scope
- System Environment includes physical, programmatic, acquisition, political, stakeholder

Table 4 – Generic complexity metric GQM model

	System entropy model
Goal	Know and project the system context
Question	What are the projected system contexts?
Entropy Metrics	p_{s1}=what is the probability that the system mission cannot be easily/accurately projected/navigated in timeframe t? p_{s2}=What is the probability that the system environment cannot be easily/accurately projected/navigated in timeframe t?
	Observer entropy model
Goal	Know and project observer capabilities
Question	What are the observer capabilities to project system context and system behaviors?
Entropy Metrics	p_{o1}=What is the probability that the observer cannot easily/accurately project/navigate the system mission in timeframe t? p_{o2}=What is the probability that the observer cannot easily/accurately project/navigate the system environment in timeframe t?
	Behavior entropy model
Goal	Know and project system behaviors
Question	What are the projected system behaviors?
Entropy Metrics	p_{b1}= What is the probability that the systems behaviors for the actual mission cannot be easily/accurately projected/navigated in timeframe t? p_{b2}= What is the probability that the systems behaviors for the actual environment cannot be easily/accurately projected/navigated in timeframe t?

The metrics are intended to be generic responses to the questions, but adaptations may be applied for different fields of study. Similarly, the generic metric categories are provided with the expectation that further field/system/program-specific derivations will be made. For example, aspects of the mission and environment can be independent questions instead of being grouped. Adding questions on uses, scale, scope, physical, programmatic, acquisition, political, stakeholder, etc. results in creation of independent metrics. See Table 5 for a summary of the steps in the complexity-based risk determination methodology.

Table 5 – Complexity-based risk determination methodology

Step 1	Evaluate the Generic Complexity Metrics GQM for the specific program or case
Step 2	Determine complexity metrics for the specific program or case
Step 3	Group metrics into risk areas (as needed)
Step 4	Perform complexity-based risk evaluation using the complexity metrics

Once the complexity metrics are derived and evaluated, those that are not extremely improbable are carried forward. The metrics carried forward provide guidance for the complexity-based risk creation. Complexity-based risks are compiled, merged as necessary, and evaluated for probability of occurrence. Evaluation of the probabilities can be approached using methods already used in systems engineering risk management. A log based scale may be preferable for evaluating lower likelihood conditions. Both linear and log scales are shown in Figure 2.

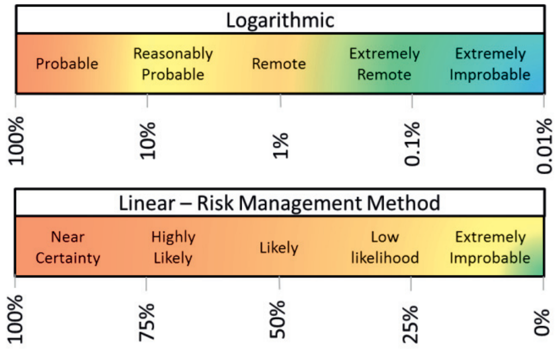

Figure 2 – Probability of occurrence definition

3.3. Case study

Next, this investigation uses the proposed methodology to create complexity-based risks for a sample Air Traffic Control (ATC) radar system. First start with a set of requirements and a sample set of risks created using standard risk management processes. The DoD generated a set of requirements for its next generation transportable ATC radar system in 2012 [18]. There are more than 800 requirements, so key performance/behavior related requirements will be assessed for risks in this example. See Table 6 for the selected requirements and associated risks. Evaluate the Probability of Failure (P_f) using standard risk management estimation technique. Programmatic risks associated with similar types of programs have been listed.

Table 6 – Program and requirements based risk assessment

Requirement	Risk	P_f
<<1>> The PSR and SSR shall detect and process a minimum of 700 real aircraft targets in any mix of PSR only, PSR/SSR merge, or SSR only targets, in the presence of up to an additional 300 false PSR reports and 100 false SSR reports (due to False-Replies Unsynchronized-in-Time (FRUIT), synchronous garble, or multipath) uniformly or non-uniformly distributed in azimuth for a 360 degree scan, and not be impacted by weather channel processing. (T=O)	If the data processor cannot support the required load, then processing capabilities updates will impact cost estimate.	10%
<<6>> Target signals received at the PSR and the SSR antenna shall be processed and displayed on all D-RAPCON ATC controller displays in less than 2.2 seconds from the time of signal reception at the PSR or SSR antenna receive ports. (T=O)	If architecture fails latency test, then architecture modifications will delay first delivery.	20%
<<23>> In the clear, the PSR shall have a single scan Probability of detection (Pd) ⩾ 80% (T), > 90% (O), for a one square meter RCS, Swerling-1 type target, within the detection volume as defined herein section 3.2.2.3, PSR Detection Probability Volume at a Probability of False Alarm (PFA) ⩽10-6 over 92 percent of the target radial velocities between -700 and +700 knots.	If the radar fails the Pd acceptance test the system budgets will need to be reevaluated impacting the delivery schedule.	10%
<<36>> Under normal clutter conditions the PSR shall report targets prior to scan-to-scan correlation with a maximum of 100 false search reports per scan in three consecutive scans.	If the radar fails the P_{fa} acceptance test the system budgets will need to be reevaluated impacting the delivery schedule.	10%
<<37>> The PSR shall detect, categorize, and report precipitation within the six reflectivity levels (18 dBZ to 57 dBZ) defined by the National Weather Service (NWS). (T=O)	Assessed as 'no risk'	-
Program risks		**P_f**
If modifications to the existing transportable radar product delay manufacturing, then liquidated damages of $TBD/week will be incurred		10%
If system fails environmental qualification testing, cost will be incurred for redesign/re-sourcing and subsequent requalification testing.		10%

The risks created using requirements as a guide are potentially useful in the context of the program and a sample solution. Next, apply the proposed complexity-based risk technique using the complexity aspects indicated in Table 7. Table 7 is a metrics summary table (derived from Table 4). It provides a down-selection of the aspects of the mission and environment applied to the sample system to communicate exactly what is being evaluated. For this example, let the timeframe '*t*' include the duration of the contract and several years of deployment.

Table 7 – Aspects of sample metrics selected for application to the case study

Model	Mission	Environment
System	What is the probability that ________ result in a system that cannot be easily/accurately projected/navigated in timeframe *t*?	
	$\mathbf{p_{s1}}$: aspects of the system requirements, uses scale/scope	$\mathbf{p_{s2}}$: development environment, political, programmatic, acquisition, stakeholder
Observer	What is the probability that the observer cannot easily/accurately project/navigate _______ in timeframe *t*?	
	$\mathbf{p_{o1}}$: subsystem/interface performance	$\mathbf{p_{o2}}$: technology supportability
Behavior	What is the probability that _________ cannot be easily/accurately projected/navigated in timeframe *t*?	
	$\mathbf{p_{b1}}$: failures during manufacturing, selloff, testing, fielding; repair/replace behaviors	$\mathbf{p_{b2}}$: obsolescence behaviors; behavior while adding new customer features/capabilities

Next, complexity metrics are compiled in Table 8 for the sample system. In cases where a risk evaluation is desirable p>EI is listed as 'yes.' Extremely Improbable (EI) indicates an area where the evaluation identifies no significant risk. The entropy model in Table 8 is the realization of Table 5, Step 2 for the case study. Metrics p_{o2}, p_{s2}, and p_{b2} introduce risks related to the future environments.

Table 8 –Complexity metrics for the case study

System entropy model			
#	**Metric**	**p>EI?**	**Detail**
p_{s1}	What is the probability that the requirements cannot be easily/accurately projected/navigated in timeframe t?	Yes	Similar result to the standard evaluation in Table 6.
p_{s2}	What is the probability that the uses/scope/physical environment cannot be easily/accurately projected/navigated in timeframe t?	Yes	Radar system may be deployed for military efforts, humanitarian efforts, and future endeavors of military TBD. Deployed landscape changes (legal but encroaching spectrum uses, windmills, technology available to create purposeful interference)
p_{s3}	What is the probability that the scale cannot be easily/accurately projected/navigated in timeframe t?	No	Past systems of this type have been moderately scaled. Planning done together with industry and academia
p_{s4}	What is the probability that the acquisition environment cannot be easily/accurately projected/navigated in timeframe t?	No	Strong customer relationship.
p_{s5}	What is the probability that the political environment cannot be easily/accurately projected/navigated in timeframe t?	No	There is political pressure to replace ATC radars with ADSB; expect military to continue use of radar in timeframe '*t*'
p_{s6}	What is the probability that the stakeholder environment cannot be easily/accurately projected/navigated in timeframe t?	Yes	Government and military-industrial staffing instability in timeframe '*t*' drives potential stakeholder instability

Observer entropy model			
#	**Metric**	**p>EI?**	**Detail**
p_{o1}	What is the probability that the observer cannot easily/accurately project/navigate the requirements in timeframe t?	No	Requirements are at the contractual stage. Changes must be negotiated and agreeable by seller/buyer.
p_{o2}	What is the probability that the observer cannot easily/accurately project/navigate the uses/scope in timeframe t?	Yes	The observer cannot accurately identify the future uses and deployment environments of the system
p_{o3}	What is the probability that the observer cannot easily/accurately project/navigate the scale in timeframe t?	No	Observer cannot predict, but planning will done together with industry and academia and changes must be negotiated and agreeable by seller/buyer.
p_{o4}	What is the probability that the observer cannot easily/accurately project/navigate the acquisition environment in timeframe t?	Yes	Observer cannot predict, due to government instability and turnover.
p_{o5}	What is the probability that the observer cannot easily/accurately project/navigate the political environment in timeframe t?	Yes	Observer cannot predict, due to government instability and turnover.
p_{o6}	What is the probability that the observer cannot easily/accurately project/navigate the stakeholder environment in timeframe t?	Yes	Observer cannot predict, due to government instability and turnover.
Behavior entropy model			
#	**Metric**	**p>EI?**	**Detail**
p_{b1}	What is the probability that the system behaviors cannot be easily/accurately projected/navigated for the *actual* system requirements in timeframe t?	No	Requirements are at the contractual stage. Changes must be negotiated and agreeable by seller/buyer.
p_{b2}	What is the probability that the system behaviors cannot be easily/accurately projected/navigated for the *actual* system uses/scope/physical environment in timeframe *t*?	Yes	The system behaviors would be impacted by variable uses/environment of the radar system which may evolve greatly over timeframe '*t*'
p_{b3}	What is the probability that the system behaviors cannot be easily/accurately projected/navigated for the *actual* system scale in timeframe *t*?	Yes	Scaling the system (power, aperture) will not make the behavior unpredictable. Scaling to add features could make the system unpredictable, but any scaling must be negotiated and will be planned together with industry and academia
p_{b4}	What is the probability that the system behaviors cannot be easily/accurately projected/navigated for the *actual* system acquisition environment in timeframe *t*?	No	Three past failures of prior transportable ATC program places pressure on presiding acquisition office to succeed on this attempt.
p_{b5}	What is the probability that the system behaviors cannot be easily/accurately projected/navigated for the *actual* system political environment in timeframe *t*?	No	Political environment will not impact system behavior
p_{b6}	What is the probability that the system behaviors cannot be easily/accurately projected/navigated for the *actual* system stakeholder environment in timeframe *t*?	No	Stakeholder environment will not impact system behavior

Note that the ATC system example was selected since it was presumed to be complex. A complex system must be selected for evaluation since there is no point trying to find complexity-based risks for a simple system. A system can be complicated, with many moving parts, but the outcome of operation can usually be understood and predicted. Complex systems may be predictable in that they have been designed to continuously adjust, but not predictable in that the system reacts to a host of stimuli. Air traffic control systems are selected as complex because the systems have to adapt to a host of inputs like weather, time of day, ground traffic, air traffic, biologic clutter, etc. resulting in different system behaviors. Stimuli that don't happen very often make it hard for the observer to evaluate a complex system because they do not happen frequently enough for us to understand how the system will respond.

Applying the reference definition to an ATC system, we would have to consider it to be moderately complex. Even though it is adapting, all of the adaptations and states of the system are designed and well understood. The area that the radar is operating in, however, generally covers more than 10,000 square miles and is used for many other purposes other than landing aircraft. Consequently, there are cases when the system will need to operate in conditions (potentially outside of requirements) that result in behavior that cannot be accurately predicted. In general, unintended consequences are an effect of complex system (which supports the reference definition of complex systems). One example is when wind farms are installed in direct line-of-site of an existing air traffic control system. Depending on the wind speed and direction, the blades of the windmill are spinning and producing significant radar returns which can result in false aircraft detections and masking of aircraft flying over the wind farms (which can cover very significant areas). The observer in the United States will have information that a wind farm is being installed and the installation and operation can be planned accordingly. The DRAPCON radar system example is transportable and may be installed in countries which do not regulate windmill installation. Similarly, they may be installed in countries where the environment is known, but bordering on other nations (within the radar coverage) which have unknown present/future environments. Not being informed of a wind-farm installation in advance may result in a failure to predict system behavior. Even the informed observer would have difficulty accurately predicting the system behavior if it is the first time the system will encounter this type of environment.

Another example is the air traffic control system operating spectrum. In areas of the world where air traffic control radar operation can extend into territories of other nations, there can easily be disagreement over which frequencies are used for which purposes. When the air traffic control radar is pointed in the direction of a nation which is using that same frequency spectrum for another purpose, the output of the air traffic control system becomes difficult to predict. The characteristics of the interfering signal are likely unknown.

The next step in the methodology, as indicated by Table 5, Step 3, is to drive risk creation and evaluation of consequences using the entropy model in Table 8. Further evaluation is only performed for metrics with p > EI. Metrics are grouped into risk areas, and the significance of the consequences determines whether introducing a new risk is justified. Table 9 contains the new complexity-based risk areas. The probability is evaluated using the linear risk management method (Figure 2). The new risks introduced in Table 9 augment the risks in Table 6.

Table 9 –Complexity-based risk evaluation

Risk	Basis	Complexity-metric based risks	P_f
C1	p_{s1}	No new risk needed: coincides with the requirements evaluation in Table 6	-
C2	$p_{s2}, p_{o2}, p_{b2}, p_{b3}$	If radar is deployed where there is conflicting spectrum use, system performance may degrade below requirements.	40%
C3	p_{s2}, p_{b2}, p_{b3}	If radar is deployed where there is conflicting coverage volume use, like windmills, system performance may degrade below requirements.	20%
C4	$p_{s6}, p_{o4}, p_{o5}, p_{o6}$	If instable stakeholder environment results in loss of key customer allies, then sell-off schedule may be extended.	30%

4. Discussion of results

This effort proposed a complexity-based risk evaluation framework to augment the traditional risk-management process. This approach is different from prior methods which do not include a guided complexity-based framework. Questions are developed based on 'goals' which come from a reference definition of complexity. The metrics are then used for complexity-based risk development. A case study was described and used to generated complexity-based metrics and complexity-based risks. While this preliminary work presented a relatively small-system example, it is hoped that in practical environments use of this framework will reveal additional area of risk not exposed by traditional methods even in very large systems and systems of systems.

The mitigation of the complexity-based risks is forward-thinking and presents the opportunity to shape the future product and the customer perception of product quality. For example, risk C2 and C3 could result in potential warrantee issues if the customer does not understand why the performance is not meeting expectations. Even if the risks are not realized until after the warrantee expires, the customer may perceive a low product quality impacting stakeholder relationships. Exposing these risks gives the opportunity to set customer expectations or even offer/implement design improvements if the customer is amenable.

With respect to the individual or team generating complexity-based risks, expertise is critical. Similar to traditional risk management, visibility into areas difficult to predict is challenging and should be assigned to the most experienced individuals. Selecting the wrong focus can prevent us from evaluating other factors – which may be the critical factors [5]. Also worthy of discussion is the expectation that expert customer and development team complexity-based risk evaluations may produce different results. Different teams (observers) may have better visibility into different aspects of the system or its behavior. This is not a problem with the method, but is indicative of the desirability for convergence of understanding between the teams. Use of the method proposed herein will help expose any different viewpoints for resolution.

5. Conclusions

Augmenting program risk management processes by adding complexity-based risk management evaluation can help expose risks that would have otherwise been neglected. This effort proposed a framework for complexity-based risk evaluation and demonstrated its use by way of an example system. A limitation of complexity-based risk evaluation in general is that as a system becomes more complex, we simply cannot fully understand all of the knowledge and ignorance measures. The reference definition of complexity as "the degree of difficulty in accurately predicting future behavior" indicates that there will be inherent inaccuracies in predicting the behavior of systems with higher complexity. Since the difficulty of accurately predicting future behavior increases with complexity (by our definition), the possibility that we may not recognize all complexity-based risks in advance increases as the complexity of a system increases. What this method proposes is that we can best identify the system complexity-based risks using expertly determined knowledge and ignorance measures.

Future research will include evolution of the framework to improve usability. There is also a need to develop impact analysis and mitigation strategies for complexity-based risks. Future case studies will include different kinds of candidate systems and a vetting of the research with industry input.

References

[1] S. R. A. M. W. B. G. W. Roshanak Nilchiani, "Quantitative Risk - Phase 1 Interim Technical Report," Systems Engineering Research Center, Hoboken, 2013.

[2] J. W. a. B. Heydari, "Complexity: Definition and Reduction Techniques," in *Complex Systems Design and Management Conference*, Paris, 2014.

[3] H. Sillitto, "On Systems Architects and Systems Architecting: Some Thoughts on Explaining The Art and Science of System Architecting," in *Proceedings of the 21st Annual of the International Council on Systems Engineering (INCOSE) International Symposium*, Singapore, 20-23 July 2009.

[4] P. Erdi, Complexity Explained, Berlin: Springer-Verlag, 2008.

[5] G. a. M. R. Sargut, "Learning To Live with Complexity," *Harvard Business Review,* pp. 68-76, September 2011.

[6] ESD Symposium Committee, "ESD Terms and Definitions Version 16," Massachusetts Institute of Technology Engineering Systems Division, Cambridge, February 2007.

[7] R. a. D. R. Dodder, "Complex Adaptive Systems and Complexity Theory: Inter-related Knowledge Domains," ESD.83: Research Seminar in Engineering Systems Massachusetts Institute of Technology, Cambridge, 2000.

[8] a. D. A. M. Sarah A. Sheard, "A Complexity Typology for Systems Engineering," Stevens Institute of Technology, Hoboken, 2010.

[9] R. Stevens, *Engineering Enterprise Systems: Challenges and Prospects,* McLean, Virginia: The MITRE Corporation, 2006.

[10] R. M. D. D. a. J. S. Douglas Stuart, "Complexity and Adaptability," DARPA META II CODA, September 2011.

[11] D. A. e. a. K. Willcox, "Stochastic Process Decision Methods for Complex Cyber-Physical Systems," Massachusets Institute of Technology, Cambridge, MA, July 2011.

[12] A. P. e. a. Brian Murray, "Complex Systems design and Analysis (CODA) Final Report," United Technologies Corporation, August 2011.

[13] J. Y. a. M. C. e. all, "Complexity Anaylsis of the Cost Effectiveness of PI-Led NASA Science Missions," *IEEE,* 2013.

[14] N. P. Suh, "Designing-in of Quality Through Axiomatic Design," *IEEE Transactions on Reliability,* p. Vol 44. No. 2, June 1995.

[15] S. B. H. R. L. Z. Ritesh Khire, "Assessing performance uncertainty in complex hybrid systems," United Technologies Research Center, East Hartford, 2010.

[16] L. M. L. a. M. C. Brennan, Software Measurement and Estimation: A Practical Approach, Hoboken: John Wiley and Sons, 2006.

[17] G. C. H. D. R. V. Basili, "THE GOAL QUESTION METRIC APPROACH," University Of Maryland, College Park, 1994.

[18] D.-R. P. Office, *Deployable Radar Approach Control System Requirements Document,* Hanscom Air Force Base (AFB), MA, 5 March 2012.